Solution : Practical Assignment - 2
Sub: Database Technology in India

[1] Create Following Tables.

Employee Table

The employee table stores information about the employees in a company.

emp_id (Primary Key)	Unique identifier for each employee.
first_name	VARCHAR(50)
last_name	VARCHAR(50)
position	VARCHAR(50)
salary	DECIMAL(10,2)
department	VARCHAR(50)
hire_date	DATE

Product Table

The product table stores information about the products that the company offers.

Column Name	Data Type
product_id	INT (Primary Key)
product_name	VARCHAR(100)
category	VARCHAR(50)
price	DECIMAL(10,2)
stock_quantity	INT
emp_id	INT (Foreign Key)
created_date	DATE

Perform following.

1. Insert 10 Records in each table.
2. Increase the salary by 5% for employees who are from ‘Purchase department’.
   - UPDATE Employee SET salary = salary * 0.05 WHERE department = 'Purchase';
3. Select all products whose prices are less than Rs.100.
   - SELECT * FROM Product WHERE price < 100;
4. List all employees in the 'Sales' department, sorted by hire_date.
   - SELECT * FROM Employee WHERE department = 'Sales' ORDER BY hire_date;
5. List all employees who sell ‘TV’ products.
   - SELECT DISTINCT e.* FROM Employee e JOIN Product p ON e.emp_id = p.emp_id WHERE p.product_name = 'TV';

6. Delete employee data whose stock of any item is less than 50.
   - DELETE FROM Employee WHERE emp_id IN (SELECT DISTINCT emp_id FROM Product WHERE stock_quantity < 50);
7. List employees whose salary is higher than the average salary of their department.
   - SELECT e.* FROM Employee e WHERE e.salary > (SELECT AVG(salary) FROM Employee WHERE department = e.department);

[2] Create the following table with necessary constraints.

Table Name: Tbl_Book_Issue

PriamryKey: B_id

Fields	Data types	Size
B_id	Number	5
B_title	Varchar2	25
Author_name	Varchar2	25
ISBN	Number	5
Category	Varchar2	25
Issue_Date	Date	15
Return_Date	Date	15

1. Enter 5 Records.
2. Display all books whose author name is “Swami Vivekanand”.
   - SELECT *FROM Tbl_Book_Issue WHERE Author_name = 'Swami Vivekanand';
3. Display the books list which issue between 02/05/2022 to 05/10/2022.
   - SELECT * FROM Tbl_Book_Issue WHERE Issue_Date BETWEEN TO_DATE('02-05-2022','DD-MM-YYYY') AND TO_DATE('05-10-2022','DD-MM-YYYY');
4. Display author name who publishes more than 5 books.
   - SELECT * FROM Tbl_Book_Issue WHERE Issue_Date BETWEEN TO_DATE('02-05-2022','DD-MM-YYYY') AND TO_DATE('05-10-2022','DD-MM-YYYY');
5. Add new column book_price after authour _name.
   - SELECT Author_name, COUNT(*) AS Total_Books FROM Tbl_Book_Issue GROUP BY Author_name HAVING COUNT(*) > 5;
6. Update column name book_category instead of category.
   - ALTER TABLE Tbl_Book_Issue ADD (book_price NUMBER(8,2));

[3] Create Following Tables.

Students Table

The students table stores information about students.

Column Name	Data Type	Description
student_id	INT (Primary Key)	Unique identifier for each student.
first_name	VARCHAR(50)	First name of the student.
last_name	VARCHAR(50)	Last name of the student.
date_of_birth	DATE	Date of birth of the student.
gender	CHAR(1)	Gender of the student (M for male, F for female).
class	VARCHAR(10)	Class or grade of the student (e.g., "BCA").
admission_date	DATE	Date of admission into the school.

Marks Table

The marks table stores information about marks scored by students in various subjects and exams.

Column Name	Data Type	Description
mark_id	INT (Primary Key)	Unique identifier for each mark entry.
student_id	INT (Foreign Key)	Reference to the student in the students table.
subject	VARCHAR(50)	Subject name (e.g., "Oracle").
exam_type	VARCHAR(50)	Type of exam (e.g., "Midterm", "Final").
marks_obtained	DECIMAL(5, 2)	Marks obtained by the student in the exam.
total_marks	DECIMAL(5, 2)	Total possible marks in the exam.
exam_date	DATE	Date of the exam.

1. Insert 10 Records in each table.
2. List all students in class "BCA", ordered by their last name.

- SELECT * FROM Students WHERE class = 'BCA' ORDER BY last_name;

3. Calculate the average marks obtained by each student.

- SELECT s.student_id, s.first_name, s.last_name, AVG(m.marks_obtained) AS average_marks FROM Students s JOIN Marks m ON s.student_id = m.student_id GROUP BY s.student_id, s.first_name, s.last_name;

4. Find the total marks obtained by each student in the "Final" exam.

- SELECT s.student_id, s.first_name, s.last_name, SUM(m.marks_obtained) AS total_final_marks FROM Students s JOIN Marks m ON s.student_id = .student_id WHERE m.exam_type = 'Final' GROUP BY s.student_id, s.first_name, s.last_name;

5. List each student's full name along with their marks in "Oracle".

- SELECT s.first_name || ' ' || s.last_name AS full_name, m.subject, m.exam_type, m.marks_obtained FROM Students s JOIN Marks m ON s.student_id = s.student_id WHERE m.subject = 'Oracle';

[4] Create following tables and perform following queries.

Account_Holder

Cust_id	Numeric(5)	Primary Key
Cust_name	Varchar2(25)	Not Null
Address	Varchar2(30)	
City	Varchar2(15)	
Account_no	Numeric(10)	Foreign Key
Ac_open_date	Date	
Mobile	Varchar2(10)	

Account_Master

Account_no	Numeric(10)	Primary Key
Ac_type	Char(1)	Set 'S' or 'C'
Tran_no	Numeric(20)	
Balance	Numeric(20)	

Perform following queries....

1. Insert 5 records in both tables.
2. Display the Records whose account type is saving.
- SELECT ah.*, am.Ac_type FROM Account_Holder ah JOIN Account_Master am ON ah.Account_no = am.Account_no WHERE am.Ac_type = 'S';
3. Insert new fields DOB in Account Holder table.
- ALTER TABLE Account_Holder ADD DOB DATE;
4. Display the Customer Name whose name start with 'J'.
- SELECT Cust_name FROM Account_Holder WHERE Cust_name LIKE 'J%';
5. Delete the Records whose city name is Mumbai.
- DELETE FROM Account_Holder WHERE City = 'Mumbai';
6. Display all records in descending order on cust_id field form account holder table.
- SELECT * FROM Account_Holder ORDER BY Cust_id DESC;
7. Display cust_name, account_no and balance.
- SELECT ah.Cust_name, ah.Account_no, am.Balance FROM Account_Holder ah JOIN Account_Master am ON ah.Account_no = am.Account_no;

[5] Create Following Tables.

Customer Table

The Customer table stores information about the bank's customers.

Column Name	Data Type	Description
customer_id	INT (Primary Key)	Unique identifier for each customer.
first_name	VARCHAR(50)	First name of the customer.
last_name	VARCHAR(50)	Last name of the customer.
date_of_birth	DATE	Date of birth of the customer.
gender	CHAR(1)	Gender of the customer (M for male, F for female).
address	VARCHAR(100)	Address of the customer.
phone_number	VARCHAR(15)	Phone number of the customer.
email	VARCHAR(50)	Email address of the customer.

Bank Account Table

The BankAccount table stores information about customer bank accounts. Each account is linked to a customer.

Column Name	Data Type	Description
account_id	INT (Primary Key)	Unique identifier for each bank account.
customer_id	INT (Foreign Key)	Reference to the customer in the Customer table.
account_type	VARCHAR(20)	Type of the account (e.g., "Current", "Savings").
balance	DECIMAL(12, 2)	Current balance of the account.
opened_date	DATE	Date when the account was opened.
status	VARCHAR(20)	Status of the account (e.g., "Active", "Closed").

1. Insert 10 Records in each table.
2. List all customers whose last name is "Patel", sorted by first name in descending order.
   - SELECT * FROM Customer WHERE last_name = 'Patel' ORDER BY first_name DESC;
3. Calculate the total balance for each account type.
   - SELECT account_type, SUM(balance) AS total_balance FROM BankAccount GROUP BY account_type;
4. Find the number of accounts for each status (e.g., Active, Closed).
   - SELECT status, COUNT(*) AS total_accounts FROM BankAccount GROUP BY status;

5. Update the status of all bank accounts with a balance below Rs.1000 and status is "Inactive".

- UPDATE BankAccount SET status = 'Closed' WHERE balance < 1000 AND status = 'Inactive';

6. Find the maximum balance for each type of account.

- SELECT account_type, MAX(balance) AS max_balance FROM BankAccount GROUP BY account_type;

[6] Create the following table with proper format and perform the following queries.

Table Name : Account_master

Field Name	Datatype	Size	Description
Acc_no	Number	3	Account Number
Acc_name	Varchar2	30	Account Holder Name
DOB	Date		Date of Birth
Opening Date	Date		Account opening date
Balance	Number	11,2	Current balance
Acc_type	Char	1	S For Saving, C for current
Gender	Char	1	M for male, F for female
Acc_status	Char	1	O for open , C for Close

Perform the following queries.

1) Insert 10 records as needed.
2) Display all information of all account holders.

- SELECT * FROM Account_master;

3) Display all closed account holder in ascending order.

- SELECT * FROM Account_master WHERE Acc_status = 'C' ORDER BY Acc_name ASC;

4) Display information of all female account holders who born before 2000.

- SELECT * FROM Account_master WHERE Gender = 'F' AND DOB < TO_DATE('2000-01-01', 'YYYY-MM-DD');

5) Count total number of Male account holder.

- SELECT COUNT(*) AS Total_Male_Account_Holders FROM Account_master WHERE Gender = 'M';

6) Delete all close account.

- DELETE FROM Account_master WHERE Acc_status = 'C';

7) Add Rs. 500 in balance that have below 2000 balance.

- UPDATE Account_master SET Balance = Balance + 500 WHERE Balance < 2000;

[7] Create Following Tables.

Facebook Table

The Facebook table stores information about each user on the social media platform.

Column Name	Data Type	Description
user_id	NUMBER (Primary Key)	Unique identifier for each user.
username	VARCHAR2(50)	Username of the Facebook user.
email	VARCHAR2(100)	Email address of the user.
date_of_birth	DATE	Date of birth of the user.
join_date	DATE	Date when the user joined Facebook.
location	VARCHAR2(100)	Location or city where the user resides.

Post Table

The Post table stores information about each post made by users on Facebook.

Column Name	Data Type	Description
post_id	NUMBER (Primary Key)	Unique identifier for each post.
user_id	NUMBER (Foreign Key)	Reference to the Facebook table for user details.
content	TEXT	Text content of the post.
post_date	DATE	Date when the post was created.
likes	NUMBER	Number of likes the post received.
shares	NUMBER	Number of times the post was shared.

CREATE TABLE Facebook (user_id NUMBER PRIMARY KEY, username VARCHAR2(50), email VARCHAR2(100), date_of_birth DATE, join_date DATE, location VARCHAR2(100));	CREATE TABLE Post (post_id NUMBER PRIMARY KEY, user_id NUMBER, content CLOB, post_date DATE, likes NUMBER, shares NUMBER, FOREIGN KEY (user_id) REFERENCES Facebook(user_id));

1. Enter 10 Records.
2. List all users located in "Bhavnagar", sorted by their join date.
- SELECT * FROM Facebook WHERE location = 'Bhavnagar' ORDER BY join_date;
3. Count the total number of posts made by each user.
- SELECT f.username, COUNT(p.post_id) AS total_posts FROM Facebook f LEFT JOIN Post p ON f.user_id = p.user_id GROUP BY f.username;
4. Find the average number of likes per post for each user.
- SELECT f.username, ROUND(AVG(p.likes), 2) AS avg_likes FROM Facebook f JOIN Post p ON f.user_id = p.user_id GROUP BY f.username;
5. Find the most-liked post for each user.
- SELECT f.username, p.post_id, p.content, p.likes FROM Post p JOIN Facebook f ON f.user_id = p.user_id WHERE p.likes = (SELECT MAX(likes) FROM Post WHERE user_id = f.user_id);
6. Update the location for a specific user (user_id = 102) to "Bhavnagar".
- UPDATE Facebook SET location = 'Bhavnagar' WHERE user_id = 102;
7. Remove a column shares from post table.
- ALTER TABLE Post DROP COLUMN shares;

[8] Create following tables in oracle with given constraints.

Table name: STUDENT

Column Name	Data Type And Size	Constraints
Sid	Number(3)	Primary Key
Name	Varchar2(20)	Not Null
City	Varchar2(20)	
Pincode	Number(6)	Unique

Table name: EXAM

Column Name	Data Type And Size	Constraints
Eid	Number(3)	
Subname	Varchar2(20)	
Marks	Number(2)	
Sid	Number(3)	Foreign Key

Insert following data

Sid	Name	City	Pincode
1	Vijay	Bhavnagar	364002
2	Ashok	Bhavnagar	364002
3	Ujaas	Rajkot	360055
4	Ajay	Bhavnagar	364002
5	Pinakin	Jamnagar	361003
6	Bhumi	Bhavnagar	364002
7	Rajul	Rajkot	360055

Eid	Subname	Marks	Sid
101	ORACLE	40	1
101	JAVA	23	2
101	PHP	24	3
101	ASP	45	1
101	SE	56	1
102	JAVA	40	2
102	PHP	34	3

Perform the Following Queries

1) Display all information of the entire STUDENT table.
   - SELECT * FROM STUDENT;
2) Display all details of EXAM table.
   - SELECT * FROM EXAM;
3) List Student Name, Subject and Marks of all the students whose EID is 101.
   - SELECT s.Name, e.Subname, e.Marks FROM STUDENT s JOIN EXAM e ON s.Sid = e.Sid WHERE e.Eid = 101;
4) List the students who never appeared any exam.
   - SELECT s.Name FROM STUDENT s WHERE s.Sid NOT IN (SELECT DISTINCT Sid FROM EXAM);
5) Count total number of students appeared in the exam 101 and 102 subjects wise.
   - SELECT Eid, Subname, COUNT(DISTINCT Sid) AS Total_Students FROM EXAM WHERE Eid IN (101, 102) GROUP BY Eid, Subname;

[9] Create Following Tables.

Account_table:

Account_no	Name	DOB	Salary	Gender
0011	Rohan	10-Mar-2001	10000	Male
0022	Utsav	12-Aprl-2001	15000	Male
0033	Greeva	12-Aug-2002	20000	Female
0044	Aarvi	12-May-2022	25000	Female
0055	Misri	05-June-2001	30000	Female
0066	Shreya	18-June-2002	15000	Female
0077	Raj	09-Sept-2003	45000	Male

Perform following.

1) Display only Female account holder records in descending order by their name.
   - SELECT * FROM Account_table WHERE Gender = 'Female' ORDER BY Name DESC;
2) Display account holder detail that has maximum salary (using function).
   - SELECT * FROM Account_table WHERE Salary = (SELECT MAX(Salary) FROM Account_table);
3) Display the name of account holder who born between 2001-2003.
   - SELECT Name, DOB FROM Account_table WHERE DOB BETWEEN TO_DATE('01-Jan-2001', 'DD-Mon-YYYY') AND TO_DATE('31-Dec-2003', 'DD-Mon-YYYY');
4) Display all account holder name in uppercase only (using function).
   - SELECT UPPER(Name) AS Uppercase_Name FROM Account_table;
5) Delete the records whose salary is 15000 only.
   - DELETE FROM Account_table WHERE Salary = 15000;
6) Update only male account holder data, increase their salary by 5%
   - UPDATE Account_table SET Salary = Salary * 1.05 WHERE Gender = 'Male';
7) Display all details of account holder whose name starts with letter ‘R’ or U’ .
   - SELECT * FROM Account_table WHERE Name LIKE 'R%' OR Name LIKE 'U%';
8) Change field name from ‘Salary’ to ‘Gross_salary’
   - ALTER TABLE Account_table RENAME COLUMN Salary TO Gross_salary;

[10] Create Following tables with appropriate data type and size, define constraint as per given instruction.

Table Name: Student

Name	Type(Size)	Constraint
Roll no.	Number(7)	Primary key
Name	Varchar2(30)	Not null
Address	Varchar2(50)	
City	Varchar2(20)	Only allowed 'Bhavnagar', 'Rajkot' and 'Surat'
Pincode	Number(6)	
Mobileno	Number (10)	
Gender	Char(1)	Must be 'M' or 'F'
DOB	Date	Not Null

1) Insert minimum 10 records.
2) Display Rollno, name and city of all students.
   - SELECT Roll_no, Name, City FROM Student;
3) Display name of students in uppercase who are living in Rajkot.
   - SELECT UPPER(Name) AS Name_Rajkot FROM Student WHERE City = 'Rajkot';
4) Display the Students whose birthday is coming in current month.
   - SELECT Name, DOB FROM Student WHERE EXTRACT(MONTH FROM DOB) = EXTRACT(MONTH FROM SYSDATE);
5) Display number of Male and Female students living in each city.
   - SELECT City, Gender, COUNT(*) AS Total_Students FROM Student GROUP BY City, Gender;
6) Display Student data whose name start with 'd'.
   - SELECT * FROM Student WHERE Name LIKE 'd%';
7) Display rollno between 3 to 5.
   - SELECT * FROM Student WHERE Roll_no BETWEEN 3 AND 5;